APPENDIX A

~~Integrated Circuit I/O Using~~
~~A High Performance Bus Interface~~
CONTROLLER DEVICE AND METHOD FOR OPERATING SAME

APPENDIX B

CROSS-REFERENCE TO RELATED APPLICATIONS

This patent application is a continuation of U.S. Patent Application No. 10/037,171, filed December 21, 2001 (pending); which is a continuation of U.S. Patent Application No. 09/835,263, filed April 13, 2001 (pending); which is a continuation of U.S. Patent Application No. 09/545,648, filed April 10, 2000 (now U.S. Patent No. 6,378,020); which is a continuation of U.S. Patent Application No. 09/161,090, filed September 25, 1998 (now U.S. Patent No. 6,049,846); which is a division of U.S. Patent Application No. 08/798,520, filed February 10, 1997 (now U.S. Patent No. 5,841,580); which is a division of U.S. Patent Application No. 08/448,657, filed May 14, 1995 (now U.S. Patent No. 5,638,334); which is a division of U.S. Patent Application No. 08/222,646, filed March 31, 1994 (now U.S. Patent No. 5,513,327); which is a continuation of U.S. Patent Application No. 07/954,945, filed September 30, 1992 (now U.S. Patent No. 5,319,755); which is a continuation of U.S. Patent Application No. 07/510,898, filed April 18, 1990 (now abandoned).

APPENDIX C

Prior art memory systems have attempted to solve the problem of high speed access to memory with limited success. U.S. Patent No. 3,821,715 (Hoff et. al.), was issued to Intel Corporation for the earliest 4-bit [~~micro-processor~~] microprocessor. That patent describes a bus connecting a single central processing unit (CPU) with multiple RAMs and ROMs. That bus multiplexes addresses and data over a 4-bit wide bus and uses point-to-point control signals to select particular RAMs or ROMs. The access time is fixed and only a single processing element is permitted. There is no block-mode type of operation, and most important, not all of the interface signals between the devices are bused (the ROM and RAM control lines and the RAM select lines are point-to-point).

APPENDIX D

In U.S. Patent No. 4,646,270[~~9~~] (Voss), a video RAM is described which implements a parallel-load, serial-out shift register on the output of a DRAM. This generally allows greatly improved bandwidth (and has been extended to 2, 4 and greater width shift-out paths.) The rest of the interfaces to the DRAM (RAS, CAS, multiplexed address, etc.) remain the same as for conventional DRAMS.

APPENDIX E

Figures 7A and 7B show[~~s~~] the timing whereby signals from two devices can overlap temporarily and drive the bus at the same time.

APPENDIX F

Figures 8A and 8B show[s] the connection and timing between bus clocks and devices on the bus.

APPENDIX G

Figure 16 is a block diagram representation of a set of internal registers within each device illustrated in Figure 2.

APPENDIX H

[~~Each~~] With reference to Figure 16, each semiconductor device contains a set of internal registers 170, preferably including a device identification (device ID) register 171, a device-type descriptor register 174, control registers 175 and other registers containing other information relevant to that type of device. In a preferred implementation, semiconductor devices connected to the bus contain registers 172 which specify the memory addresses contained within that device and access-time registers 173 which store a set of one or more delay times at which the device can or should be available to send or receive data.

Most of these registers can be modified and preferably are set as part of an initialization sequence that occurs when the system is powered up or reset. During the initialization sequence each device on the bus is assigned a unique device ID number, which is stored in the device ID register 171. A bus master can then use these device ID numbers to access and set appropriate registers in other devices, including access-time registers 173, control registers 175, and memory registers 172, to configure the system. Each slave may have one or several access-time registers 173 (four in a preferred embodiment). In a preferred embodiment, one access-time register in each slave

is permanently or semi-permanently programmed with a fixed value to facilitate certain control functions. A preferred implementation of an initialization sequence is described below in more detail.

APPENDIX I

The data block transfer occurs later at a time specified in the request packet control information, preferably beginning on an even cycle. A device begins a data block transfer almost immediately with a device-internal phase as the device initiates certain functions, such as setting up memory addressing, before the bus access phase beings. The time after which a data block is driven onto the bus lines is selected from values stored in slave access-time registers 173. The timing of data for reads and writes is preferably the same; the only difference is which device drives the bus. For reads, the slave drives the bus and the master latches the values from the bus. For writes the master drives the bus and the selected slave latches the values from the bus.

APPENDIX J

Slave devices do not need to detect a collision directly, but they must wait to do anything irrecoverable until the last byte (byte 5) is read to ensure that the packet is valid. A request packet with Master[0:3] equal to 0 (a retry signal) is ignored and does not cause a collision. The subsequent bytes of such a packet are ignored.

APPENDIX K

In the bus-based system of this invention, a mechanism is provided to give each device on the a unique device identifier (device ID) after power-up or under other conditions as desired or needed by the system. A master can then use this device ID to access a specific device, particularly to set or modify registers 170 of the specified device, including the control and address registers. In the preferred embodiment, one master is assigned to carry out the entire system configuration process. The master provides a series of unique device ID numbers for each unique device connected to the bus system. In the preferred embodiment, each device connected to the bus contains a special device-type register which specifies the type of device, for instance CPU, 4 MBit memory, 64 MBit memory or disk controller. The configuration master should check each device, determine the device type and set appropriate control registers, including access-time registers 173. The configuration master should check each memory device and set all appropriate memory address registers 172.

One means to set up unique device ID numbers is to have each device to select a device ID in sequence and store the value in an internal device ID register 171. For example, a master can pass sequential device ID numbers through shift

registers in each of a series of devices, or pass a token from device to device whereby the device with the token reads in device ID information from another line or lines. In a preferred embodiment, device ID numbers are assigned to devices according to their physical relationship, for instance, their order along the bus.

APPENDIX L

The configuration master should choose and set an access time in each access-time register 173 in each slave to a period sufficiently along to allow the slave to perform an actual, desired memory access. For example, for a normal DRAM access, this time must be longer than the row address strobe (RAS) success time. If this condition is not met, the slave may not deliver the correct data. The value stored in a slave access-time register 173 is preferably one-half the number of bus cycles for which the slave device should wait before using the bus in response to a request. Thus an access time value of '1' would indicate that the slave should not access the bus until at least two cycles after the last byte of the request packet has been received. The value of AccessReg0 is preferably fixed at 8 (cycles) to facilitate access to control registers.

APPENDIX M

In a preferred embodiment, a standard data block size can be selected for use with ECC, and the ECC method will determine the required number of bits of information in a corresponding ECC block. RAMs containing ECC information can be programmed to store an access time that is equal to: (1) the access time of the normal RAM (containing data) plus the time to access a standard data block (for corrected data) minus the time to send a request packet (6 bytes); or[,] (2) the access time of a normal RAM minus the time to access a standard ECC block minus the time to send a request packet. To read a data block and the corresponding ECC block, the master simply issues a request for the data immediately followed by a request for the ECC block. The ECC RAM will wait for the selected access time then drive its data onto the bus right after (in case (1) above)) the data RAM has finished driving out the data block. Persons skilled in the art will recognize that the access time described in case 920 above can be used to drive ECC data before the data is driven onto the bus lines and will recognize that writing data can be done by analogy with the method described for a read. Persons skilled in the art will also recognize the adjustments that must be made in the bus-busy structure and the request packet arbitration methods of this invention in order to

accommodate these paired ECC response.

APPENDIX N

Referring to [~~Fig. 7~~]Figures 7A and 7B, although there is no stable condition where two devices drive the bus at the same time, conditions can arise because of propagation delay on the wires where one device, A 41, can start driving its part of the bus 44 while the bus is still being driven by another device, B 42 (already asserting a logical 1 on the bus). In a system using current drivers, when B 42 is driving the bus (before time 46), the value at points 44 and 45 is logical 1. If B 42 switches off at time 46 just when A 41 switches on, the additional drive by device A 41 causes the voltage at the output 44 of A 41 to drop briefly below the normal value. The voltage returns to its normal value at time 47 when the effect of device B 42 turning off is felt. The voltage at point 45 goes to logical 0 when device B 42 turns off, then drops at time 47 when the effect of device A 41 turning on is felt. Since the logical 1 driven by current from device A 41 is propagated irrespective of the previous value on the bus, the value on the bus is guaranteed to settle after one time of flight (t_f) delay, that is, the time it takes a signal to propagate from one end of the bus to the other. If a voltage drive was used (as in ECL wired-ORing), a logical 1 on the bus (from device B 42 being previously driven) would prevent the transition put out by

device A 41 being felt at the most remote part of the system, e.g., device 43, until the turnoff waveform from device B 42 reached device A 41 plus one time of flight delay, giving a worst case settling time of twice the time of flight delay.

APPENDIX O

Clocking a high speed bus accurately without introducing error due to propagation delays can be implemented by having each device monitor two bus clock signals and then derive internally a device clock, the true system clock. The bus clock information can be sent on one or two lines to provide a mechanism for each bused device to generate an internal device clock with zero skew relative to all the other device clocks. Referring to Figure 8A, in the preferred implementation, a bus clock generator 50 at one end of the bus propagates an early bus clock signal in one direction along the bus, for example on line 53 from [~~left to~~]right to left, to the far end of the bus. The same clock signal then is passed through the direct connection shown to a second line 54, and returns as a late bus clock signal along the bus from the far end to the origin, propagating from [~~right to~~]left to right. A single bus clock line can be used if it is left unterminated at the far end of the bus, allowing the early bus clock signal to reflect back along the same line as a late bus clock signal.

APPENDIX P

Referring to Figure 9, each primary bus unit can be mounted on a single circuit board 66, sometimes called a memory stick. Each transceiver device 19 in turn connects to a transceiver bus 65, similar or identical in electrical and other respects to the primary bus 18 described at length above. In a preferred implementation, all masters are situated on the transceiver bus so there are no transceiver delays between masters and all memory devices are on primary bus units so that all memory access experience an equivalent transceiver delay, but persons skilled in the art will recognize how to implement systems which have masters on more than one bus unit and memory devices on the transceiver bus as well as on primary bus units. In general, each teaching of this invention which refers to a memory device can be practiced using a transceiver device and one or more memory devices on an attached primary bus unit. Other devices, generically referred to as peripheral devices, including disk controllers, video controllers or I/O devices can also be attached to either the transceiver bus or a primary bus unit, as desired. Persons skilled in the art will recognize how to use a single primary bus unit or multiple primary bus units as needed with a transceiver bus in certain system designs.

APPENDIX Q

A block diagram of the preferred input/output circuit for address/data/control lines is shown in Figure 10. This circuitry is particularly well-suited for use in DRAM devices but it can be used or modified by one skilled in the art for use in other devices connected to the bus of this invention. It consists of a set of input receivers 71, 72 and output driver 76 connected to input/output line 69 and 75 and circuitry to use the internal clock 73 and internal clock complement 74 to drive the input interface. The clocked input receivers take advantage of the synchronous nature of the bus. To further reduce the performance requirements for device input receivers, each device pin, and thus each bus line, is connected to two clocked receivers, one to sample the even cycle inputs, the other to sample the odd cycle inputs. By thus de-multiplexing the input [~~70~~]69 at the pin, each clocked amplifier is given a full 2 ns cycle to amplify the bus low-voltage-swing signal into a full value CMOS logic signal. Persons skilled in the art will recognize that additional clocked input receivers can be used within the teachings of this invention. For example, four input receivers could be connected to each device pin and clocked by a modified internal device clock to transfer sequential bits from the bus to internal device circuits, allowing still higher

external bus speeds or still longer settling times to amplify the bus low-voltage-swing signal into a full value CMOS logic signal.

APPENDIX R

The input receivers of every slave must be able to operate during every cycle to determine whether the signal on the bus is a valid request packet. This requirement leads to a number of constraints on the input circuitry. In addition to requiring small acquisition and resolution delays, the circuits must take little or no DC power, little AC power and inject very little current back into the input or reference lines. The standard clocked DRAM sense amp shown in Figure 11 satisfies all these requirements except the need for low input currents. When this sense amp goes from sense to sample, the capacitance of the internal nodes 83 and 84 in Figure_11 is discharged through the reference line 68 and input 69, respectively. This particular current is small, but the sum of such currents from all the inputs into the reference lines summed over all devices can be reasonably large.

APPENDIX S

In the preferred embodiment, two sets of these delay lines are used, one to generate the true value of the internal device clock 73, and the other to generate the complement 74 without adding any inverter delay. The dual circuit allows generation of truly complementary clocks, with extremely small skew. The complement internal device clock is used to clock the 'even' input receivers to sample at time 127, while the true internal device clock is used to clock the 'odd' input receivers to sample at time 125. The true and complement internal device clocks 73 and 74, respectively, are also used to select which data is driven to the output drivers. The gate delay between the internal device clock and output circuits driving the bus is slightly greatly than the output circuits driving the bus is slightly greater than the corresponding delay for the input circuits, which means that the new data always will be driven on the bus slightly after the old data has been sampled.

APPENDIX T

Running the internal I/O lines in the conventional way at high bus cycle rates is not possible. In the preferred method, several (preferably 4) bytes are read or written during each cycle and the column access path is modified to run at lower rate (the inverse of the number of bytes accessed per cycle, preferably ¼ of the bus cycle rate). Three different techniques are used to provide the additional internal I/O lines required and to supply data to memory cells at this rate. First, the number of I/O bit lines in each subarray running through the column decoder 147A and 147B is increased, for example, to 16, eight for each of the two columns of column sense amps and the column decoder selects one set of columns from the "top" half 148 of subarray 150 and one set of columns from the "bottom" half 149 during each cycle, where the column decoder selects one column sense amp per I/O bit line. Second, each column I/O line is divided into two halves, carrying data independently over separate internal I/O lines from the left half 147A and right half 147B of each subarray (dividing each subarray into quadrants) and the column decoder selects sense amps from each right and left half of the subarray, doubling the number of bits available at each cycle. Thus each column decode selection turns on *n* column sense amps, where *n* equals four (top left and

right, bottom left and right quadrants) times the number of I/O lines in the bus to each subarray quadrant (8 lines each x 4 = 32 lines in the preferred implementation). Finally, during each RAS cycle, two different subarrays, e.g. 157 and 153, are accessed. This doubles again the available number of I/O lines containing data. Taken together, these changes increase the internal I/O bandwidth by at least a factor of 8. Four internal buses are used to route these internal I/O lines. Increasing the number of I/O lines and then splitting them in the middle greatly reduces the capacitance of each internal I/O line which in turn reduces the column access time, increasing the column access bandwidth even further.

APPENDIX U

1-150 (cancelled).

151 (New). A method of operating a controller device, comprising:

outputting a value to a memory device;

outputting a first operation code to the memory device, wherein the first operation code instructs the memory device to store the value in a register of the memory device;

outputting a block size value to the memory device, wherein the block size value indicates an amount of read data to be output by the memory device in response to a second operation code;

outputting the second operation code to the memory device, wherein the second operation code instructs the memory device to perform a read operation, wherein the second operation code includes precharge information that indicates whether the memory device should precharge sense amplifiers on the memory device after sensing data corresponding to the read operation; and

after a read delay following the outputting of the second operation code, sampling a first portion of the read data output by the memory device in response to the second operation code, wherein the read delay is determined using the value output to

the memory device for storage in the register.

152 (New). The method of claim 151, wherein outputting the first and second operation codes further comprises outputting the first and second operation codes synchronously with respect to a clock signal.

153 (New). The method of claim 151, wherein outputting the first operation code further comprises outputting the first operation code using pads on the controller device, the pads to connect to a set of external signal lines, and wherein outputting the second operation code further comprises outputting the second operation code using the pads on the controller device used to output the first operation code.

154 (New). The method of claim 151, wherein sampling the first portion of the read data further comprises:

for a pad on the controller device from which read data is sampled, sampling two bits of read data from the pad during a clock cycle of a clock signal used by the controller device.

155 (New). The method of claim 154, wherein sampling the first portion of the read data further comprises:

for each pad on the controller device from which read data is sampled, sampling two bits of read data during a clock cycle of a clock signal used by the controller device.

156 (New). The method of claim 154, further comprising generating the clock signal internal to the controller device.

157 (New). The method of claim 156, wherein generating the clock signal further comprises:

receiving first and second external clock signals; and

generating the clock signal using the first and second external clock signals.

158 (New). The method of claim 154, further comprising receiving the clock signal from external to the controller device.

159 (New). The method of claim 154, further comprising:

outputting a third operation code to the memory device, wherein the third operation code instructs the memory device to perform a write operation; and

after a write delay following the outputting of the third operation code, outputting write data to the memory device,

wherein the write delay is determined using the value output to the memory device for storage in the register.

160 (New). The method of claim 159, wherein the write data is output synchronously with respect to the clock signal.

161 (New). The method of claim 160, wherein the second operation code is output synchronously with respect to the clock signal.

162 (New). The method of claim 161, further comprising:

outputting first address information to the memory device, wherein the first address information indicates a location in a memory array of the memory device at which the read data is stored; and

outputting second address information to the memory device, wherein the second address information indicates a location in the memory array at which the write data is to be stored.

163 (New). The method of claim 162, wherein outputting the write data to the memory device further comprises, for a pad on the controller device used to output write data, outputting two bits of write data using the pad during a clock cycle of the

clock signal.

164 (New). The method of claim 163, wherein outputting the write data to the memory device further comprises, for each pad on the controller device used to output write data, outputting two bits of write data during a clock cycle of the clock signal.

165 (New). The method of claim 163, wherein outputting the value further comprises outputting the value to the memory device via an external bus, wherein the first address information and the second address information are also output to the memory device via the external bus.

166 (New). The method of claim 165, wherein the read delay and the write delay are about the same.

167 (New). The method of claim 166, wherein outputting the block size value and outputting the second operation code further comprises outputting the block size value and the second operation code to the memory device in a packet.

168 (New). The method of claim 151, further comprising sampling the amount of read data output by the memory device in

response to the second operation code, wherein the amount of read data is sampled over a plurality of clock cycles of a clock signal used by the controller device to output the first and second operation codes, and wherein sampling the amount of read data includes sampling the first portion of the read data.

169 (New). A controller device, comprising:

output driver circuitry, the output driver circuitry to:

output a value to a memory device;

output a first operation code to the memory device, wherein the first operation code instructs the memory device to store the value in a register in the memory device;

output a block size value to the memory device, wherein the block size value indicates an amount of read data to be output by the memory device in response to a second operation code; and

output the second operation code to the memory device, wherein the second operation code instructs the memory device to perform a read operation, wherein the second operation code includes precharge information that indicates to the memory device whether the memory device should precharge sense amplifiers on the memory device after sensing data corresponding to the read operation; and

input receiver circuitry to sample a first portion of the read data output by the memory device in response to the second operation code, the input receiver circuitry to sample the first portion of the read data after a read delay following the outputting of the second operation code, wherein the read delay is determined using the value output to the memory device for storage in the register.

170 (New). The controller device of claim 169, further comprising a plurality of pads to interface with signal lines external to the controller device, wherein the input receiver circuitry is coupled to a first portion of the plurality of pads to receive the first portion of the read data, and wherein for a pad used in the receipt of the first portion of the read data, two bits of read data are received from the pad during a clock cycle of a clock signal used by the controller device.

171 (New). The controller device of claim 170, wherein for each pad used in the receipt of the first portion of the read data, two bits of read data are received during a clock cycle of the clock signal.

172 (New). The controller device of claim 170, further

comprising a clock receiver to receive the clock signal from external to the controller device.

173 (New). The controller device of claim 170, further comprising a clock generation circuit to generate the clock signal.

174 (New). The controller device of claim 173, further comprising:

a first clock receiver to receive a first external clock signal; and

a second clock receiver to receive a second external clock signal,

wherein the clock generation circuit is coupled to the first and second clock receivers, and wherein the clock generation circuit generates the clock signal using the first and second external clock signals.

175 (New). The controller device of claim 170, wherein the input receiver circuitry includes, for each pad of the plurality of pads used to receive read data, a first input receiver and a second input receiver.

176 (New). The controller device of claim 169, wherein the output driver circuitry outputs a third operation code to the memory device, wherein the third operation code instructs the memory device to perform a write operation.

177 (New). The controller device of claim 176, wherein the output driver circuitry outputs write data to the memory device after a write delay following outputting of the third operation code, wherein the write delay is determined using the value output to the memory device for storage in the register.

178 (New). The controller device of claim 177, wherein the output driver circuitry includes a plurality of output drivers, wherein the output driver circuitry outputs at least a portion of the write data using output drivers of the plurality of output drivers that are also used to output the first operation code to the memory device.

179 (New). The controller device of claim 177, wherein the output driver circuitry includes a plurality of output drivers, wherein an output driver of the plurality of output drivers outputs two bits of the write data during a clock cycle of a clock signal used by the controller device.

180 (New). The controller device of claim 179, further comprising a clock receiver to receive the clock signal, wherein the output driver circuitry outputs the write data synchronously with respect to the clock signal.

181 (New). The controller device of claim 169, wherein the output driver circuitry includes:

a first set of output drivers to output the first operation code; and

a second set of output drivers to output the block size value.

182 (New). A method of controlling a memory device by a controller device, comprising:

outputting a value to the memory device;

outputting a first operation code to the memory device, wherein the first operation code instructs the memory device to store the value in a register of the memory device;

outputting a second operation code to the memory device, wherein the second operation code instructs the memory device to perform a read operation, wherein the second operation code includes precharge information that indicates whether the memory

device should precharge sense amplifiers on the memory device after sensing data corresponding to the read operation; and

after a read delay following the outputting of the second operation code, sampling a first portion of read data output by the memory device in response to the second operation code, wherein the read delay is determined using the value output to the memory device for storage in the register.

183 (New). The method of claim 182, wherein sampling the first portion of read data further comprises:

for each pad on the controller device from which read data is sampled, sampling two bits of read data during a clock cycle of a clock signal used by the controller device.

184 (New). The method of claim 183, further comprising:

outputting a third operation code to the memory device, wherein the third operation code instructs the memory device to perform a write operation; and

after a write delay following the outputting of the third operation code, outputting write data to the memory device, wherein the write delay is determined using the value output to the memory device for storage in the register, wherein outputting the write data to the memory device further

comprises, for each pad on the controller device used to output write data, outputting two bits of write data during a clock cycle of the clock signal.

185 (New). A controller device, comprising:

means for outputting a value to the memory device;

means for outputting a first operation code to the memory device, wherein the first operation code instructs the memory device to store the value in a register of the memory device;

means for outputting a block size value to the memory device, wherein the block size value indicates an amount of read data to be output by the memory device in response to a second operation code;

means for outputting the second operation code to the memory device, wherein the second operation code instructs the memory device to perform a read operation, wherein the second operation code includes precharge information that indicates whether the memory device should precharge sense amplifiers on the memory device after sensing data corresponding to the read operation; and

means for sampling read data, wherein after a read delay following the outputting of the second operation code, the means for sampling read data samples a first portion of the read data

output by the memory device in response to the second operation code, wherein the read delay is determined using the value output to the memory device for storage in the register.

186 (New). The controller device of claim 185, further comprising:

for each pad on the controller device from which read data is sampled, means for sampling two bits of read data during a clock cycle of a clock signal used by the controller device.

187 (New). The controller device of claim 186, further comprising:

means for outputting a third operation code to the memory device, wherein the third operation code instructs the memory device to perform a write operation; and

means for outputting write data to the memory device, wherein after a write delay following the outputting of the third operation code, the means for outputting write data to the memory device outputs write data to the memory device, wherein the write delay is determined using the value output to the memory device for storage in the register, wherein outputting the write data to the memory device further comprises, for each pad on the controller device used to output write data,

outputting two bits of write data during a clock cycle of the clock signal.

188. A controller device, comprising:

output driver circuitry, the output driver circuitry to:

output a value;

output a first operation code, wherein the first operation code represents an instruction to a memory device to store the value in a register in the memory device;

output a block size value, wherein the block size value indicates an amount of read data to be output by the memory device in response to a second operation code; and

output the second operation code, wherein the second operation code represents an instruction to the memory device to perform a read operation, wherein the second operation code includes an indication to the memory device as to whether the memory device should precharge sense amplifiers on the memory device after sensing data corresponding to the read operation; and

input receiver circuitry to sample a first portion of the read data output by the memory device, the input receiver circuitry to sample the first portion of the read data after a read delay following the outputting of the second operation

code, wherein the read delay is determined using the value.

APPENDIX V

~~The present invention includes a memory subsystem comprising at least two semiconductor devices, including at least one memory device, connected to a bus, where the bus includes a plurality of bus lines for carrying substantially all address, data and control information needed by said memory devices, where the control information includes device select information and the bus has substantially fewer bus lines than the number of bits in a single address, and the bus carries device select information without the need for separate device select lines connected directly to individual devices.~~

~~The present invention also includes a protocol for master and slave devices to communicate on the bus and for registers in each device to differentiate each device and allow bus requests to be directed to a single or to all devices. The present invention includes modifications to prior art devices to allow them to implement the new features of this invention. In a preferred implementation, 8 bus data lines and an AddressValid bus line carry address, data and control information for memory addresses up to 40 bids wide.~~

A controller device and method for operating same is disclosed. In one particular exemplary embodiment, the controller device may comprise output driver circuitry and input

receiver circuitry. The output driver circuitry may output a value, a first operation code, a block size value, and second operation code. The first operation code may represent an instruction to a memory device to store the value in a register in the memory device. The block size value may indicate an amount of read data to be output by the memory device in response to the second operation code. The second operation code may represent an instruction to the memory device to perform a read operation. The input receiver circuitry may sample a first portion of the read data output by the memory device after a read delay following the outputting of the second operation code.

APPENDIX W

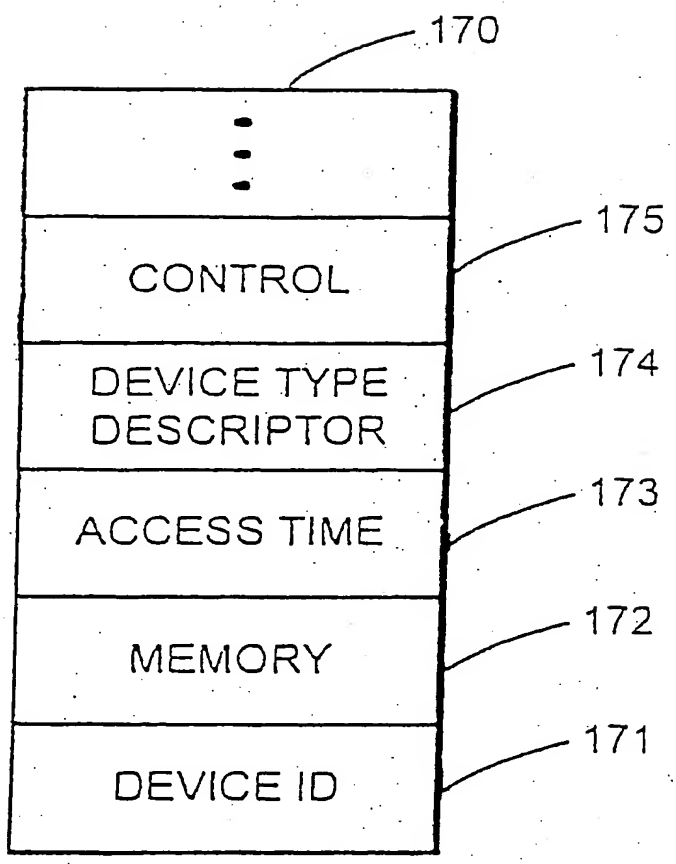
170
CONTROL
175
DEVICE TYPE DESCRIPTOR
174
ACCESS TIME
173
MEMORY
172
DEVICE ID
171

FIG. 16

OIPE
FEB 05 2004
PATENT & TRADEMARK OFFICE

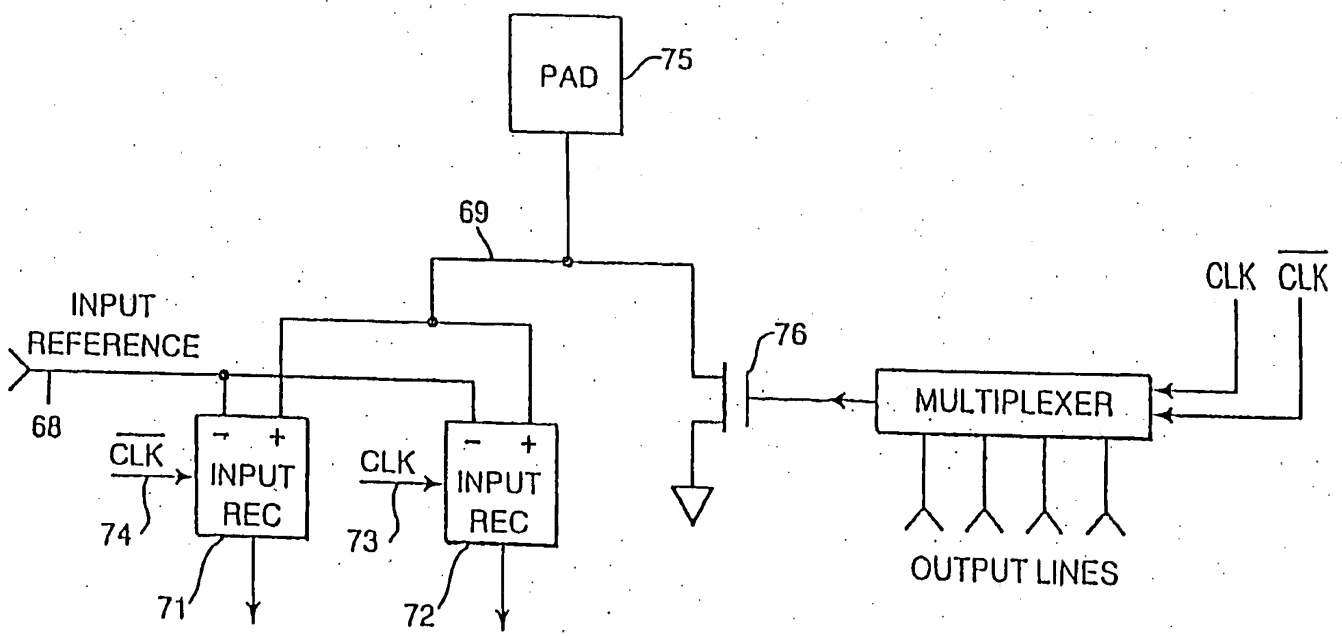

FIG_10

APPENDIX X